

client unique identifiable status, pending commands and/or data components. In this scheme, commands and signaling transmitted between the servers and clients utilize standard HTTP protocol semantics and HTML or standard markup language syntax. Clients encapsulate or embed information as parameters passed to HTTP CGI as a set of standard HTTP conversations. A CGI processing program converts, parses or processes each conversation and passes arguments with or without data to queues. [Each conversation includes a client identification key(s) and commands which are structured as attribute-value pair tuples. The service-handler client connects to the central server and accesses the client queue on the central server to check for any pending commands or update their status in either synchronous or asynchronous manner. The synchronous server access scheme is regulated temporally by either deterministic clocking on server response or by a server based adaptive algorithm which can monitor network and client activities and optimize client access patterns.]

#### **AMENDMENTS TO THE CLAIMS:**

The following listing of claims will replace all prior versions and listings of claims in the captioned Application:

#### **Listing Of Claims:**

Claim 1 (currently amended): A service broker system for interactive monitoring and control of data to and from computers and Internet enabled devices of a

client/server safety system over the Internet, for processing data from a data network including at least one data source, the system comprising:

a first communication module for initiating communication with a moderator and adapted to receive data from the moderator;

a second communication module for sending data to at least one of the data source and the moderator;

a service-action module for processing the received data and for performing a task based on the processed data; and

an export module in communication with the service-action module and for publishing data based at least in part on the performed task to the data network.

Claim 2 (currently amended) The service broker [according to] system set forth in claim 1, wherein the published data is published to at least one of the data source and the moderator.

Claim 3 (currently amended) The service broker [according to] system set forth in claim 1, wherein the moderator includes a data store and the first communication module receives data from the data store

Claim 4 (currently amended) The service broker [according to] system set forth in claim 3, wherein the data store stores data received from the at least one data source.

Claim 5 (currently amended) The service broker [according to] system set forth in claim 3, wherein the data store is a command queue and the data received from the data source is a command which is intended to be processed by the service-action module, the first communication module receiving the command from the queue.

Claim 6 (currently amended) The service broker [according to] system set forth in claim 1, wherein the first communication module communicates with the moderator via the HTTP protocol.

Claim 7 (currently amended) The service broker [according to] system set forth in claim 4, wherein the data source communicates with the moderator via the HTTP protocol.

Claim 8 (currently amended) The service broker [according to] system set forth in claim 1, wherein the performed task comprises communicating a command based at least in part on the processed data to a device connected to the service broker.

Claim 9 (currently amended) The service broker [according to] system set forth in claim 1, wherein at least one of the moderator and the data source comprises a virtual representation of the service broker and wherein the published data updates the virtual representation.

Claim 10 (currently amended) The service broker [according to] system set forth in claim 8, wherein at least one of the moderator and the data source comprises a virtual representation of the connected device and wherein the published data updates the virtual representation.

Claim 11 (currently amended) A method for transferring data from a data source to a service broker system for interactive monitoring and control of data to and from computers and Internet enabled devices of a client/server safety system over the Internet, the method comprising the steps of:

providing a data source and a service broker; [providing a moderator for receiving the data transferred by the data source; providing a data store for storing data received by the moderator; providing a communications module for transferring data from the data store; providing a virtual representation of the service broker on the data source;]

providing a moderator for receiving the data transferred by the data source;

providing a data store for storing data received by the moderator;

providing a communications module for transferring data from the data store;

providing a virtual representation of the service broker on the data source;

transferring data from the data source to the moderator, the data sent being related to or associated with the virtual representation;

storing the data received by the moderator in the data store;

retrieving the data from the data store via the communications module and forwarding the data to the service broker; and [updating the virtual representation when the

service broker receives the data sent by the data source, whereby data is transferred between the data source and to the service broker.]

updating the virtual representation when the service broker receives the data sent by the data source, whereby data is transferred between the data source and to the service broker.

Claim 12 (currently amended) The method [according to] set forth in claim 11, wherein the data transferred from the data source to the moderator is performed using the HTTP protocol.

Claim 13 (currently amended) The method [according to] set forth in claim 11, wherein the data transferred from the moderator to the service broker is performed via the HTTP protocol.

Claim 14 (currently amended) The method [according to] set forth in claim 12, wherein the data is transferred using name/value pairs.

Claim 15 (currently amended) The method [according to] set forth in claim 13, wherein the data is transferred using name/value pairs.

Claim 16 (currently amended) The method [according to] set forth in claim 14, wherein the name/value pair is transmitted using a field/value abstraction layer.

Claim 17 (currently amended) The method [according to] set forth in claim 15, wherein the name/value pair is transmitted using a field/value abstraction layer.

Claim 18 (currently amended) The method [according to] set forth in claim 11, wherein the data is a command for changing the state of the service broker and wherein the virtual representation is updated when the state of the service broker is changed.

Claim 19 (currently amended) The method [according to] set forth in claim 11, wherein the data sent is a command for changing the state of the service broker.

Claim 20 (currently amended) The method [according to] set forth in claim 11, wherein the moderator and the data store are the same entity.

Claim 21 (currently amended) The method [according to] set forth in claim 11, wherein the data store and the communications module are the same device.

Claim 22 (currently amended) The method [according to] set forth in claim 11, wherein the moderator, data store, and the communications module are the same entity.

Claim 23 (currently amended) The method [according to] set forth in claim 11, wherein there are a plurality of data stores.

Claim 24 (currently amended) The method [according to] set forth in claim 11, wherein there are a plurality of service brokers.

Claim 25 (currently amended) The method [according to] set forth in claim 11, wherein there are a plurality of moderators and data stores.

Claim 26 (currently amended) The method [according to] set forth in claim 11, wherein the data source and the service broker are at the same node.

Claim 27 (currently amended) The method [according to] set forth in claim 11, wherein the data store is a queue of commands.

Claim 28 (currently amended) The method [according to] set forth in claim 11, wherein the data transferred from the data store to the service broker is initiated by the service broker.

Claim 29 (currently amended) The method [according to] set forth in claim 11, wherein the data is retrieved by specifying a specific device identifier.

Claim 30 (currently amended) The method [according to] set forth in claim [11] 18, wherein at least one device is connected to the service broker and wherein the